



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/042,354	01/11/2002	Paul Matthew Carpenter	550-296	6802

7590 03/16/2005

NIXON & VANDERHYE P.C.
8th Floor
1100 North Glebe Rd.
Arlington, VA 22201-4714

EXAMINER

PATEL, HETUL B

ART UNIT	PAPER NUMBER
----------	--------------

2186

DATE MAILED: 03/16/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/042,354

Applicant(s)

CARPENTER ET AL.

Examiner

Hetul Patel

Art Unit

2186

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 03 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 26 January 2005.
2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-7 and 9-12 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) 1-7 and 9-12 is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☒ All b) ☐ Some * c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Specification

1. This action is responsive to appeal brief filed on January 26, 2005.
2. In view of the appeal brief filed on January 26, 2005, PROSECUTION IS HEREBY REOPENED. New grounds of rejections are set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) request reinstatement of the appeal.

If reinstatement of the appeal is requested, such request must be accompanied by a supplemental appeal brief, but no new amendments, affidavits (37 CFR 1.130, 1.131 or 1.132) or other evidence are permitted. See 37 CFR 1.193(b)(2).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claim 1 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gulley et al. (USPN: 5,025,407) hereinafter, Gulley in view of TI TMS32010 User's Guide,

hereinafter, TI, further in view of Messina et al. (USPN: 4,317,168) hereinafter, Messina and further in view of Langendorf et al. (USPN: 4,860,197) hereinafter, Langendorf.

As per claim 1, Gulley teaches a data processing apparatus comprising a main processor (the graphics processor 120 in Fig. 1) responsive to a program instruction to perform data processing operations; and a coprocessor (the floating point coprocessor 1200 in Fig. 1) coupled to the main processor. Furthermore, Gulley teaches that the coprocessor loads (accepts) one or more loaded data words (a set of operands) from the main processor. The coprocessor also performs the operation on the loaded operands according to an instruction loaded (accepted) from the main processor and provides the result to the main processor (e.g. see Col. 2, lines 3-10 and Fig. 1).

Gulley fails to clearly teach that both loading one or more data words and performing an operation to provide the result are performed in response to a single coprocessor load instruction on the main processor. However, TI, on the other hand, teaches that co-processor such as TMS32010 runs instruction called "LTD", which combines three sub instructions "LT", "APAC" and "DMOV" (e.g. see page 3-7). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to combine Gulley's two instructions, one for loading data words and second for performing an operation to provide result, into one instruction as taught by TI. In doing so, it will increase the processing speed and it will be more user/programmer friendly since the user/programmer does not need to worry about adding all the sub instructions in the program/code.

However, both Gulley and TI failed to teach that the number of loaded data words loaded into the coprocessor is depended upon whether or not the start address of the operand data is aligned with a word boundary. Messina, on the other hand, teaches that, on the main processor, the number of loaded data words (the quad words, QW) loaded (for the line fetch, LF) is depended upon the operand data alignment within the word boundary, i.e. 8 or 9 quad words (QW) occur for a line fetch (LF) depending upon the double word (DW) boundary alignment (e.g. see Abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the step of deciding whether to load one or more loaded data words based on the operand data alignment within the word boundary as taught by Messina in the apparatus taught by the combination of Gulley and TI. In doing so, the coprocessor load instruction gets the required number of operands and can start the execution of the load instruction without waiting for the remaining operands. Therefore, the number of clock cycles required for the execution of the coprocessor load instruction is reduced.

None of Gulley, TI or Messina teaches the further limitation of having an alignment register for storing a value specifying alignment between the operand data and the one or more loaded data words. Langendorf, on the other hand, teaches that the system includes one or more memory sets for storing alignment values which represent whether the boundary of the instruction with one or more parcels (e.g. see the abstract and claim 6). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the alignment register

for storing alignment value as taught by Langendorf in the apparatus taught by the combination of Gulley, TI and Messina so the required number of operands are loaded based on the alignment value and the execution of the load instruction is started without waiting for the remaining operands. Therefore, the number of clock cycles required for the execution of the coprocessor load instruction is reduced.

4. Claims 11 and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulley in view of TI, further in view of Messina.

As per claims 11 and 12, Gulley teaches a method of processing data and a computer program product for controlling a computer comprising the steps of: in response to program instructions performing data processing operations in a main processor (the graphics processor 120 in Fig. 1) and in response to a coprocessor load instruction (an instruction) on the main processor, a coprocessor (the floating point coprocessor 1200 in Fig. 1) loads (accepts) one or more loaded data words (a set of operands) from the main processor. The coprocessor also performs the operation on the loaded operands according to an instruction loaded (accepted) from the main processor and provides the result to the main processor (e.g. see Col. 2, lines 3-10 and Fig. 1).

Gulley fails to clearly teach that both loading one or more data words and performing an operation to provide the result are performed in response to a single coprocessor load instruction on the main processor. However, TI, on the other hand, teaches that co-processor such as TMS32010 runs instruction called "LTD", which

combines three sub instructions "LT", "APAC" and "DMOV" (e.g. see page 3-7).

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to combine Gulley's two instructions, one for loading data words and second for performing an operation to provide result, into one instruction as taught by TI. In doing so, it will increase the processing speed and it will be more user/programmer friendly since the user/programmer does not need to worry about adding all the sub instructions in the program/code.

However, both Gulley and TI failed to teach that the number of loaded data words loaded into the coprocessor is depended upon whether or not the start address of the operand data is aligned with a word boundary. Messina, on the other hand, teaches that the number of loaded data words (the quad words, QW) loaded (for the line fetch, LF) is depended upon the operand data alignment within the word boundary, i.e. 8 or 9 quad words (QW) occurred for a line fetch (LF) depending upon the double word (DW) boundary alignment (e.g. see Abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the step of deciding whether to load one or more loaded data words based on the operand data alignment within the word boundary as taught by Messina in Gulley's method and computer program. In doing so, the coprocessor load instruction gets the required number of operands and can start the execution of the load instruction without waiting for the remaining operands. Therefore, the number of clock cycles required for the execution of the coprocessor load instruction is reduced.

5. Claims 2-7 and 9 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulley in view of TI, further in view of Messina, further in view of Langendorf and further in view of York et al. (USPN: 6,002,881) hereinafter, York.

As per claim 2, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that the coprocessor includes a coprocessor memory for storing one or more locally stored data words used as operands in the at least one coprocessor processing operation in combination with the one or more loaded data words. York, on the other hand, teaches that the coprocessor (Piccono coprocessor 4 in Fig. 1) includes a coprocessor memory (registers 10 in Fig. 2) for storing one or more data words, which includes data words used as operands and loaded data words (emphasis added) (e.g. see Figs. 1-2 and Col. 5, lines 44-57). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the coprocessor memory in the coprocessor for storing locally stored data words along with the loaded data words as taught by York in the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf. In doing so, the coprocessor retrieves these data words faster than storing it elsewhere (not locally to the coprocessor), which reduces the data latency and therefore, the performance of the coprocessor increases.

As per claim 3, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that the data processing apparatus comprising a memory coupled to the main processor and wherein the main processor is configured to retrieve the one or more loaded data words

from the memory to the coprocessor via the main processor without being stored within registers within the main processor. York, on the other hand, teaches a memory coupled to the main processor and wherein the one or more loaded data words are retrieved from the memory to the coprocessor via the main processor without being stored within registers within the main processor (e.g. see Col. 1, lines 18-34).

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to modify the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf as such so the loaded data words can be retrieved from the memory to the coprocessor via the main processor without being stored within registers within the main processor as taught by York. In doing so, the data retrieval time reduces and therefore, the overall performance of the data processing apparatus increases.

As per claim 4, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that the main processor includes a register operable to store an address value pointing to the one or more data words. York, on the other hand, teaches that the main processor (the CPU) includes a register, which holds an address value pointing to the data words (e.g. see Col. 2, lines 42-51). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to modify the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf by adding an register in the main processor for storing an address value as taught by York so the start address within the memory to be accessed is determined.

As per claims 5 and 6, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that at least one coprocessor processing operation includes calculating a sum of absolute differences between a plurality of byte values within the one or more loaded data words and corresponding ones of a plurality of byte values within the one or more locally stored data words. York, on the other hand, teaches that one of the coprocessor processing operation (the SUBA instruction) calculates sum of differences (e.g. see Col. 36, lines 55-58). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to modify the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf so the SUBA instruction can be run as taught by York. In doing so, the sum of differences between byte values within loaded and stored data words is calculated for the correlation purposes.

As per claim 7, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that the sum of absolute differences is accumulated within an accumulate register of the coprocessor. York, on the other hand, teaches that sum of differences that calculated by the SUBA instruction is accumulated (added) in an accumulate register (the third register) (e.g. see Col. 36, lines 55-58). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to modify the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf so the sum of differences is accumulated in the accumulate register as

taught by York. In doing so, the sum of differences can be retrieved anytime by the coprocessor for any required manipulation. Since it is stored locally in the coprocessor register, coprocessor can retrieve it quickly compare to if it is stored elsewhere.

As per claim 9, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that the coprocessor load instruction includes an offset value to be added to the address value upon execution. York, on the other hand, teaches that the offset value (offset field within the instruction) is used by the CPU to specify the changes to be made in the address value provided by the CPU upon execution of a particular instruction (e.g. see the abstract and Col. 2, lines 42-51). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to modify the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf so upon the execution of an instruction, an offset that included in the instruction is added to the address value as taught by York. In doing so, the actual address is calculated from the given address value by adding an offset to that given address.

6. Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gulley in view of TI, further in view of Messina, further in view of Langendorf and further in view of Wu et al. (USPN: 6,418,166) hereinafter, Wu.

As per claim 10, the combination of Gulley, TI, Messina and Langendorf teaches the claimed invention as described above. However, none of them teaches that at least one coprocessor processing operation calculates a sum of absolute differences as part

of block pixel value matching. Wu, on the other hand, teaches that the sum of differences is used as the search criteria in the block matching process (e.g. see Fig. 8 and Col. 4, lines 42-44). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to use the sum of absolute differences as a part of block pixel value matching as taught by Wu in the data processing apparatus taught by the combination of Gulley, TI, Messina and Langendorf. In doing so, it finds a block of pixels that most closely matches the source block of pixels. Therefore, it is advantageous.

Remarks

7. As to the remark, Applicant asserted:

- (a) Gulley does not teach that a load instruction used to load the operand data into the coprocessor also specifies the operation to be performed by the coprocessor on the loaded operand data.
- (b) Messina does not relate to loading data words into a coprocessor.
- (c) Messina does not load a variable number of words into a cache. As taught in Col. 4, line 45 – Col. 5, line 10, Messina reads a variable number of QWs from a main memory but writes a **fixed** (not variable) number of DWs into the cache.
- (d) There would have been no motivation to combine the teachings of Gulley and Messina and the Examiner's motivation is improperly based on hindsight.

- (e) There would have been no motivation to combine the teachings of Messina and Langendorf because Langendorf's teachings do not add to or improve upon Messina's teachings.
- (f) The alignment values of Langendorf do not serve as a trigger to specify a variable number of data words to be loaded into a coprocessor as claimed.

Examiner respectfully traverses Applicant's remark for the following reasons:

With respect to (a), as described above in the rejection of claim 1, Gulley teaches that the coprocessor loads (accepts) one or more loaded data words (a set of operands) from the main processor. The coprocessor also performs the operation on the loaded operands according to an instruction loaded (accepted) from the main processor and provides the result to the main processor (e.g. see Col. 2, lines 3-10 and Fig. 1). The Examiner agreed with the Applicant that Gulley does not clearly teach that a load instruction used to load the operand data into the coprocessor also specifies the operation to be performed by the coprocessor on the loaded operand data. However, TI, on the other hand, teaches that co-processor such as TMS32010 runs instruction called "LTD", which combines three sub instructions "LT", "APAC" and "DMOV" (e.g. see page 3-7). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to combine Gulley's two instructions, one for loading data words and second for performing an operation to provide result, into one instruction as taught by TI. In doing so, it will increase the processing speed and it

will be more user/programmer friendly since the user/programmer does not need to worry about adding all the sub instructions in the program/code.

With respect to (b), the Examiner agrees with the Applicant that Messina does not teach about loading data words into a coprocessor. However, Messina does teach about reading (loading) a variable number of quad words (QWs) from the main memory (e.g. see the abstract). And, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to use the Messina's processor as the coprocessor, i.e. to use the same technique of loading a variable number of data words into the coprocessor taught by the combination of Gulley and TI.

With respect to (c), the Examiner did not find anywhere in the cited column and lines, i.e. Col. 4, line 45 – Col. 5, line 10, that Messina writes a **fixed** (not variable) number of DWs into the cache. Therefore, this argument has been mooted.

With respect to (d) and (e), in response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992).

In response to applicant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon

Art Unit: 2186

hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

With respect to (f), Langendorf, teaches that the system includes the alignment register (one or more memory sets) for storing a value specifying alignment (alignment values) between the operand data (branch instruction) and the one or more loaded data words (data parcels) (e.g. see the abstract and claim 6).

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hetul Patel whose telephone number is 571-272-4184. The examiner can normally be reached on M-F 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt Kim can be reached on 571-272-4182. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2186

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

HBP

HBP

A handwritten signature in black ink, appearing to read 'M. Anderson', with a long horizontal flourish extending to the right.

MATTHEW D. ANDERSON
PRIMARY EXAMINER